

# Sistem Monitoring Dan Kontrol Kualitas Air Pada Kolam Ikan Koi Berbasis Internet of Things (IoT)

**Andhika Bayu Pratama, Imam Much Ibnu Subroto, Andi Riansyah**

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Sultan Agung

Correspondence Author : [andhikabayu@std.unissula.ac.id](mailto:andhikabayu@std.unissula.ac.id)

## Abstract

Saat ini ikan koi banyak diminati oleh beberapa pecinta ikan hias pada kolam rumah atau perkantoran. Melihat ikan koi berenang di dalam kolam dapat menurunkan tekanan darah & denyut jantung, selain itu juga dapat dijadikan sebagai stress relief pada tubuh manusia. Ikan koi merupakan ikan hias yang sangat sensitif terhadap kualitas air, oleh karena itu kualitas air sangat berperan penting terhadap perkembangan ikan koi itu sendiri. Tidak sedikit pengelola ikan koi yang dapat mengetahui parameter-parameter apa saja yang dibutuhkan ikan koi supaya dapat berkembang dengan baik. Oleh karena itu perlu adanya suatu sistem yang dapat memonitor dan mengontrol kualitas air pada kolam ikan koi. Sistem ini bertujuan untuk mengetahui parameter air yang dibutuhkan ikan koi sekaligus dapat mengontrol parameter air tersebut. Pada penelitian ini menggunakan metode forward chaining, yaitu merupakan metode yang dapat melakukan pencarian dari masalah yang ditunjukkan untuk mencari solusi. Hasil pengujian sistem ini adalah dengan menggunakan metode forward chaining kualitas air telah dikendalikan dengan baik dalam menentukan keputusan dan pengendalian sesuai aturan pakar. Selain itu sistem dapat menjalankan fungsi pengambilan data, pengambilan keputusan, pengiriman data ke cloud dan dapat menjalankan aktuator.

Keyword : forward chaining, ikan koi, Internet of Things (IoT)

## 1. PENDAHULUAN

Ikan hias merupakan komoditas perikanan yang banyak diminati oleh banyak orang, baik dalam maupun luar negeri. Keberadaan ikan hias di area rumah menjadi salah satu kesenangan tersendiri untuk orang yang memeliharanya. Salah satu ikan hias yang diminati oleh masyarakat adalah ikan koi. Ikan koi diminati karena memiliki keindahan bentuk dan corak warnanya, bahkan banyak orang yang percaya ikan koi dapat membawa keberuntungan bagi orang yang memeliharanya. Selain itu melihat ikan koi berenang di kolam dapat menurunkan tekanan darah dan denyut jantung, dan juga berfungsi sebagai *stress relief* atau pelepas stres. Ada beberapa faktor yang menentukan bagus tidaknya kualitas warna koi yaitu kualitas ikan koi (70%), kualitas air (20%), dan faktor lainnya (10%). Ada beberapa parameter air yang berperan dalam penentu kualitas air ikan koi yaitu pH, *Dissolved Oxygen* (DO) atau oksigen terlarut, dan suhu. Disisi lain kekeruhan air menjadi aspek penting dalam pembudidayaan ikan koi, mengingat ikan koi merupakan ikan hias yang mengharuskan air di dalam kolam jernih. Kekeruhan adalah dimana air di dalamnya terdapat sedikit atau banyak partikel-partikel yang kadang tak kasat mata sehingga diperlukan alat untuk mengukur kondisi air dengan skala NTU

Pemanfaatan teknologi saat ini menjadi solusi sebagai sarana untuk dapat meningkatkan kualitas ikan koi. Salah satunya adalah sistem "Monitoring Dan Kontrol Kualitas Air Pada Kolam Ikan Koi Berbasis *Internet Of Things* (IoT)" adalah sistem untuk memonitor dan mengontrol parameter yang diperlukan oleh ikan koi.

## 2. TINJAUAN PUSTAKA

Penelitian-penelitian terdahulu yang mirip dengan pokok pembahasan penelitian ini telah banyak dilakukan oleh peneliti-peneliti sebelumnya. Penelitian Vernanda dan kawan-kawan yang dilatar belakangi oleh kualitas air dan perawatan ikan koi mendapatkan solusi dengan memanfaatkan teknologi sebagai peningkatan kualitas ikan koi lebih baik. Pada penelitiannya menyimpulkan bahwa nilai *input* yang bersifat samar akan diubah menjadi nilai tegas berupa nilai yang kontradiksi. Hal ini tentunya akan meningkatkan keakuratan hasil proses identifikasi kualitas air. Setelah Dianalisis sistem ini telah bekerja dengan baik sesuai dengan rancangan yang dibuat yakni identifikasi kualitas air kolam ikan koi [1].

Selanjutnya pada penelitian yang dilakukan oleh Elba Lintang dan kawan-kawan yang menyimpulkan bahwa, kualitas air merupakan parameter utama dalam keberhasilan dalam memelihara ikan. Parameter tersebut antara lain kadar pH dan suhu air. Pada penelitian tersebut digunakan konfigurasi topologi *multipoint to point* dan *bus*. Elba dan Firdaus menyimpulkan bahwa akurasi perangkat pengukuran keasaman (pH) diatas 90%. Kelemahan dari sensor ini

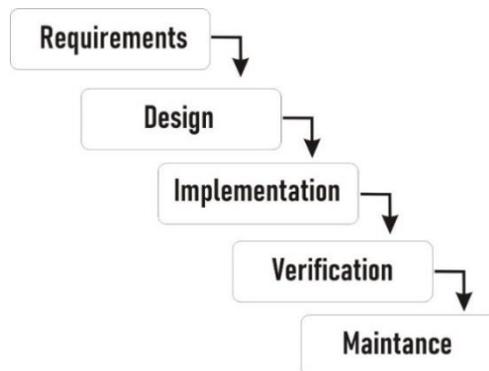
adalah masih berupa sensor analog. Apabila menggunakan sensor digital maka hasil yang diperoleh akan lebih akurat [2].

Pada penelitian Rozzef Pramana berpendapat bahwa pemantauan kualitas pada kolam budidaya banyak dilakukan secara manual sehingga membutuhkan waktu yang lama. Rozzef merancang sistem kontrol dan monitoring kualitas air pada kolam budidaya ikan berbasis web. Rozzef juga berkesimpulan bahwa perangkat mampu mengontrol perubahan suhu pada kolam secara otomatis, penurunan suhu sebesar  $0,1^{\circ}\text{C}$  pada kolam dengan kapasitas air 10 liter membutuhkan waktu 18 detik, dan untuk menurunkan suhu sebesar  $1^{\circ}\text{C}$  membutuhkan waktu 180 detik (3 menit). Untuk menaikkan suhu sebesar  $1^{\circ}\text{C}$  membutuhkan waktu 2640 detik atau 44 menit [3].

### 3. METODE PENELITIAN

#### 3.1. Pengumpulan Data

Pada tahap ini adalah proses pengumpulan data dan informasi yang ada hubungannya dengan budidaya ikan koi di dalam kolam. Untuk metode yang digunakan pada sistem ini adalah *waterfall* yang ditunjukkan pada Gambar 1

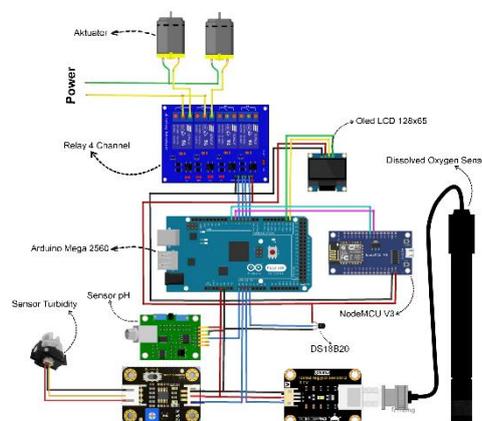


Gambar 1 Metode pengerjaan sistem

Dalam tugas akhir ini yang pertama kali dilakukan adalah analisis data di berbagai sumber, kemudian menspesifikasikan data, setelah data terspesifikasi lanjut selanjutnya adalah mendesain sistem atau *project* yang akan dibuat, koding atau menanam kode program dalam sistem, setelah pengkodean dilanjutkan dengan pengujian sistem apakah sudah dapat berjalan sesuai yang diinginkan, setelah semua sesuai dengan yang diharapkan dilanjutkan dengan implementasi sistem.

#### 3.2. Perancangan Perangkat Sistem

Dalam langkah ini ditunjukkan proses perancangan sistem, pada gambar 2 menunjukkan diagram rangkaian sistem yang menggunakan metode *Forward Chaining*. *Input* dalam sistem ini berasal dari DFRobot *Analog Turbidity Sensor*, PH Sensor, Dallas DS18B20 *Temperature Sensor*, *Analog Dissolved Oxygen Sensor*. Nantinya *input* yang diperoleh dari sensor dikirim ke *board* Arduino Mega 2560.



Gambar 2 Rangkaian sistem

Pada gambar diatas adalah bentuk dari diagram rangkaian sistem yang dihubungkan oleh komponen-komponen yang nantinya akan digunakan. Komponen tersebut terdiri dari Arduino Mega 2560 sebagai pemrosesan

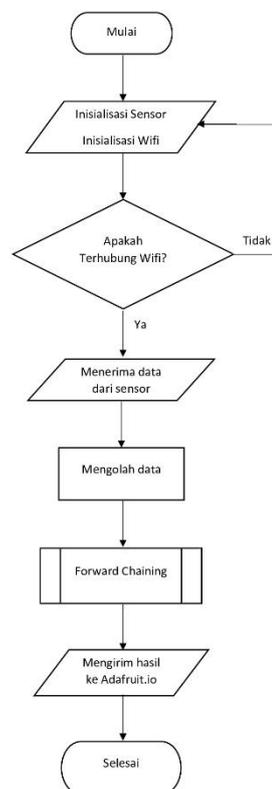
data, NodeMCU sebagai jembatan untuk pengiriman data dari board Arduino Mega 2560 ke penyedia layanan web yaitu Adafruit.io. *Input* dalam sistem ini berasal dari DFRobot *Analog Turbidity Sensor*, PH Sensor, Dallas DS18B20 *Temperature Sensor*, dan DFRobot *Analog Dissolved Oxygen Sensor*. Sedangkan untuk *output* sendiri terdapat pompa dan aerator yang nantinya pompa dan aerator mendapatkan perintah dari board Arduino Mega 2560.

### 3.3. Forward Chaining

*Forward chaining* adalah mesin inferensi *multiple* dari inferensi yang melakukan pencarian dari masalah yang ditujukan untuk mencari solusi. Sedangkan inferensi adalah proses yang ditujukan untuk menghasilkan suatu informasi dari fakta yang telah diketahui atau diasumsikan. Dalam sistem pakar, proses inferensi dilakukan dalam modul yang disebut *inference engine* (mesin inferensi). Terdapat dua metode inferensi pada sistem pakar, yaitu *forward chaining* (runut maju) dan *backward chaining* (runut mundur) [4].

### 3.4. Flowchart

*Flowchart* merupakan gambaran grafik dari setiap langkah atau urutan prosedur dalam suatu proyek. *Flowchart* ini dapat mempengaruhi penyelesaian masalah yang perlu dievaluasi lebih mendalam [5]. Berikut ini adalah langkah-langkah yang akan ditampilkan menggunakan *flowchart*. Proses ini menjelaskan bagaimana sistem mengambil data melalui sensor hingga mengirimkan data. Pada gambar 3 menampilkan proses kerja sistem



Gambar 3 Flowchart

### 3.5. Perancangan Rule

- Rule 1  
IF *Turbidity*  $\leq$  2000 OR  
PH < 8.7 OR  
PH > 7.0  
THEN Pompa Mati
- Rule 2  
IF *Turbidity* < 2000 OR  
PH > 8.7 OR  
PH < 7.0  
THEN Pompa Hidup

- *Rule 3*  
IF DO < 5.0  
THEN Aerator Hidup
- *Rule 4*  
IF DO > 5.0  
THEN Pompa Mati

#### 4. HASIL DAN PEMBAHASAN

##### 4.1. Realisasi Sistem

Pada langkah ini merupakan realisasi desain dari sistem. Terdapat wadah yang dijadikan sebagai pengganti air kolam yang ditunjukkan pada gambar 4.1 dibawah. Selain itu juga terdapat lampu sebagai pengganti pompa dan aerator. Lampu kuning untuk menggantikan aerator, sedangkan lampu putih dijadikan sebagai pengganti pompa. Ada empat sensor yang dimasukkan ke dalam air yaitu sensor DO (*Dissolved Oxygen*) atau oksigen terlarut, sensor pH, sensor *turbidity* atau kekeruhan, dan sensor suhu. Pada gambar 4 terdapat *panel box* yang di dalamnya terdapat beberapa perangkat keras yaitu arduino mega 2560, nodemcu esp8266, relay, dan pada bagian luar *panel box* terdapat oled lcd.



Gambar 4 Realisasi Desain Sistem

Selain merealisasikan desain pada sistem dilakukan konfigurasi pada sistem dengan menyetting *source code* yang dapat dilihat pada gambar 5 berikut ini.

```
Monitoring_Kolam_Koi  Sensor_DO  Sensor_PH  Sensor_Tur...
#include <SoftwareSerial.h>
SoftwareSerial s(6, 7);
String datakirim;

//===== PARAMETER =====

float kekeruhan = 2000.0;
float PHminimal = 7.0;
float PHmaksimal= 8.7;
float Oksigen = 5.0;

//===== OLED =====
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

// Declaration for an SSD1306 display connected to I2C
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire);
```

Gambar 5 *Source code* pada Arduino Mega

Pada gambar 5 diatas menunjukkan *source code* yang di *upload* ke dalam Arduino Mega 2560. Pada *source code* diatas diperlihatkan nilai pada setiap parameter. Nilai pada parameter kekeruhan diberi nama variabel kekeruhan dengan nilai 2000, yang dimana nilai 2000 adalah nilai batasan maksimum pada parameter ideal kekeruhan atau *turbidity*. Selain itu juga telah dimasukkan nilai pada parameter pH, yang dimana pada parameter pH

terdapat nilai minimum dan maksimum ideal. Pada *source code* diatas ditunjukkan nilai parameter minimum dengan nama variabel PHminimal dengan nilai 7.0 dan nilai maksimum dengan nama variabel PHmaksimal bernilai 8.7. Parameter *dissolved oxygen* (DO) diberi nama variabel Oksigen dengan nilai batasan senilai 5.0. Tipe data *float* merupakan tipe data yang digunakan untuk nilai berupa pecahan. Pada penelitian ini terdapat NodeMCU yang digunakan sebagai jembatan untuk menghubungkan data yang telah tersimpan ke adafruit.io. Berikut *source code* yang di *upload* ke dalam NodeMCU.

```
Monitoring_Kolam_Koi_NodeMCU_Adafruit_IO.ino
#include <SoftwareSerial.h>
SoftwareSerial s(D5, D6);

#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"

/***** WiFi Access Point *****/

#define WLAN_SSID      "iaesLaptop"
#define WLAN_PASS      "iaessemarang01"

/***** Adafruit.io Setup *****/

#define AIO_SERVER      "io.adafruit.com"
#define AIO_SERVERPORT  1883          // use 8883 for SSL
#define AIO_USERNAME    "Koi_Smart_Pond"
#define AIO_KEY          "aio_fcGm4lLFJUZlhwifD0gP8f5rHPik"
/***** Global State (you don't need to change this!) *****/

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// or... use WiFiClientSecure for SSL
```

Gambar 6 *Source code* pada NodeMCU

## 4.2. Uji Coba Sistem

### A. Pengujian pada kadar *Dissolved Oxygen* Rendah

Pada pengujian ini kadar air yang terdapat pada wadah merupakan air yang memiliki kadar *dissolved oxygen* rendah. Pengujian ini berfungsi apakah sensor dan aerator dapat berfungsi dengan baik. Pada gambar 4.4 memperlihatkan rangkaian sensor dan aktuator pada sistem.



Gambar 7 Pengujian kadar DO

Pada gambar 7 diatas memperlihatkan sensor yang telah dimasukkan ke dalam wadah yang berisi air. Ada empat sensor yang dimasukan ke dalam air yaitu sensor DO, pH, kekeruhan, dan suhu. Pada gambar tersebut memperlihatkan lampu kuning menyala yang artinya adalah aerator pada kolam sedang berfungsi. Untuk tampilan pada adafruit.io ditunjukkan pada gambar 8 berikut



Gambar 8 Tampilan Adafruit.io pada parameter DO rendah

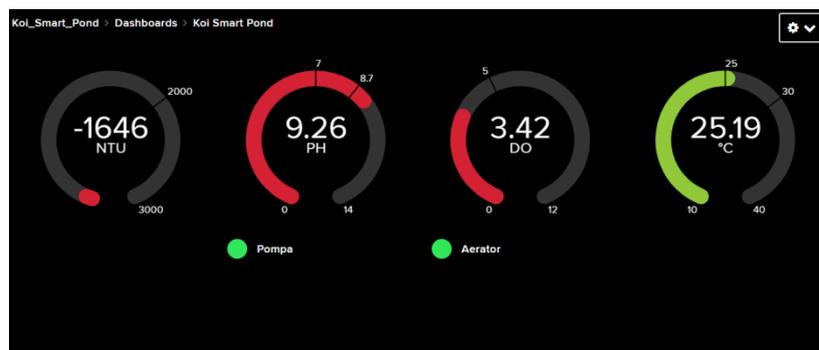
### B. Pengujian pada kadar pH tinggi dan Kadar DO rendah

Pada tahap ini yang diuji meliputi sensor pH dan sensor DO, sedangkan untuk aktuator sendiri yang diuji meliputi aerator dan pompa. Pada gambar 4.6 diperlihatkan sensor yang telah dimasukkan ke dalam air.



Gambar 9 Pengujian pada kadar pH tinggi dan DO rendah

Pada gambar 9 diatas menunjukkan bahwa sensor telah dimasukkan ke dalam air yang memiliki kadar pH tinggi dan kadar DO rendah. Pada gambar di atas menunjukkan bahwa lampu putih dan lampu kuning menyala, yang berarti pompa dan aerator sedang berfungsi. Tampilan pada adafruit.io diperlihatkan pada gambar 10 dibawah ini.



Gambar 10 Tampilan adafruit.io pada kadar pH tinggi dan DO rendah

### C. Pengujian pada Kadar Kekeruhan Tinggi

Pada tahap ini yang diuji adalah sensor kekeruhan itu sendiri dan pengujian pada pompa kolam. Gambar 4.8 menunjukkan rangkaian yang sensornya sudah dimasukkan ke dalam air.



Gambar 11 Pengujian kadar kekeruhan tinggi

Pada gambar 11 diatas menunjukkan lampu putih menyala yang diartikan pada sistem pompa sedang berfungsi. tampilan pada adafruit.io ditunjukkan pada gambar 12 dibawah



Gambar 12 Tampilan adafruit.io pada kadar kekeruhan tinggi

#### 4.3. Hasil Pengujian

Pada langkah ini data uji dan parameter yang sudah ditentukan akan dilakukan proses inferensi atau pencarian kesimpulan menggunakan metode *forward chaining* berbasis aturan. Langkah pertama yang harus dilakukan adalah melakukan pengecekan data uji terhadap basis aturan (*rule base*) dengan cara pencocokan data dengan aturan yang ada. Mulai dari sini akan ditentukan apakah proses inferensi akan dilanjutkan atau dibatalkan. Secara sederhananya yaitu apabila data uji sesuai dengan *rule* maka proses inferensi menuju ke *rule* berikutnya, dan apabila data uji tidak sesuai maka proses inferensi dibatalkan, kemudian mengalihkan ke proses inferensi dengan hasil *output* yang berbeda sampai ditemukan *rule* yang sesuai dengan data uji. Hasil inferensi ditampilkan pada tabel 1. *Source code* inferensi ditampilkan pada gambar 4.10

```

if (ntu >= kekeruhan || ph_act > PHmaksimal || ph_act < PHminimal) {
    digitalWrite(relay1, relayON);
    pompa = 1;
}
else {
    digitalWrite(relay1, relayOFF);
    pompa = 0;
}

if (sensorDo < Oksigen) {
    digitalWrite(relay2, relayON);
    aerator = 1;
}
else {
    digitalWrite(relay2, relayOFF);
    aerator = 0;
}

```

Gambar 13 Source code forward chaining

Tabel 1 Inferensi forward chaining

No	Rule	Kesimpulan
1	IF "A" OR "B" OR "C" THEN "T"	Pompa Hidup
2	IF "O" OR "P" OR "Q" THEN "U"	Pompa Mati
3	IF "D" THEN "V"	Aerator Hidup
4	IF "R" THEN "W"	Aerator Mati
5	IF "A" AND "B" AND "C" AND "D" THEN "X"	Pompa & Aerator Hidup

Keterangan :

- "A" merupakan kondisi dimana parameter kekeruhan lebih atau sama dengan 2000 NTU
- "B" merupakan kondisi dimana parameter pH lebih dari 8.7
- "C" merupakan kondisi dimana parameter pH kurang dari 7.0
- "D" merupakan kondisi dimana parameter DO kurang dari 5.0 ppm
- "O" merupakan kondisi dimana parameter kekeruhan kurang dari 2000 NTU
- "P" merupakan kondisi dimana parameter pH kurang dari 8.7
- "Q" merupakan kondisi dimana parameter pH lebih dari 7.0
- "R" merupakan kondisi dimana parameter DO lebih dari 5.0 ppm
- "T" merupakan kondisi apabila pompa hidup
- "U" merupakan kondisi apabila pompa mati
- "V" merupakan kondisi apabila aerator hidup
- "W" merupakan kondisi apabila aerator mati

#### A. Forward chaining

Pengujian *Forward Chaining* dilakukan untuk menguji sistem apakah sudah sesuai dengan kaidah-kaidah metode *Forward Chaining*. Pengujian ini dilakukan dengan pemantauan pengukur sensor dan kondisi dari pompa dan aerator. Dengan hasil pengujian terlihat pada tabel 2 berikut.

Tabel 2 Hasil Pengujian

No	Waktu	Pengukuran Sensor	Aksi	Diagnosa Pakar	Keterangan
1	18 Nov 2021	- pH : 7.6 - DO : 8.2 - Turb : - 1089 - Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
2	19 Nov 2021	- pH : 8.0 - DO : 7.6 Turb : - 900 Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
3	20 Nov 2021	- pH : 8.9 - DO : 7.1 - Turb : -1100 - Suhu : 28 °C	Pompa Menyala, Aerator Mati	Parameter pH air tinggi	Sesuai
4	22 Nov 2021	- pH : 7.7 - DO : 7.2 - Turb : -1980 - Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
5	23 Nov 2021	- pH : 7.8 - DO : 7.0 - Turb : - 1015 - Suhu : 27 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
6	24 Nov 2021	- pH : 9.1 - DO : 7.2 - Turb : -1109 - Suhu : 27 °C	Pompa menyala, Aerator Mati	Parameter pH air tinggi	Sesuai
7	25 Nov 2021	- pH : 8.1 - DO : 7.9 - Turb : - 910 - Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
8	26 Nov 2021	- pH : 8.0 - DO : 6.8 - Turb : -1112 - Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
9	27 Nov 2021	- pH : 8.1 - DO : 7.0 - Turb : -1100 - Suhu : 27 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai
10	28 Nov 2021	- pH : 7.9 - DO : 7.2 - Turb : -970 - Suhu : 28 °C	Pompa & Aerator Mati	Kondisi air kolam baik	Sesuai

### B. Pengukuran akurasi sensor

Pengukuran akurasi ini dilakukan dengan cara pengamatan di lapangan secara langsung sebanyak 10 kali. Pengukuran akurasi sistem ini sudah didapatkan nilai akurasi sebesar 100 %. Untuk menghitung nilai akurasi yaitu dengan cara menghitung data yang “benar” lalu dibagi dengan banyaknya pengukuran lalu dikali 100 %, maka akan diketahui besar akurasi metode *forward chaining* berbasis aturan dalam memberikan nilai pada sistem ini [6]. Berikut ini merupakan langkah untuk mengukur akurasi pada sistem monitoring dan kontrol kualitas air pada kolam ikan koi berbasis *internet of things* :

$$\text{Akurasi} = \frac{\text{jumlah prediksi benar}}{\text{jumlah total pengujian}} \times 100\% \quad (1)$$

$$\frac{10}{10} \times 100\% = 100\%$$

Sehingga nilai akurasi sensor yang dihasilkan adalah 100 %

## 5. KESIMPULAN

Berdasarkan hasil dari perancangan dan implementasi Sistem Monitoring Dan Kontrol Kualitas Air Pada Kolam Ikan Koi Berbasis *Internet Of Things* (IoT) dan berdasarkan pengujian maka dapat diambil kesimpulan bahwa Sistem Monitoring dan Kontrol Ikan Koi Berbasis *Internet of Things* ini dapat menjalankan fungsi pengambilan data, pengambilan keputusan, pengiriman data ke *cloud* dan menjalankan aktuator. Selain itu kualitas air telah dikendalikan dengan baik menggunakan metode *forward chaining* dalam menentukan keputusan dan pengendalian kualitas air yang optimal sesuai dengan aturan-aturan pakar dengan parameter PH, *Dissolved Oxygen* (DO), dan *Turbidity*.

## DAFTAR PUSTAKA

- [1] Verananda, Ibnu Dwi, Soewarto Hardhienata, and Arie Qur'ania. 2015. "Sistem Identifikasi Kualitas Air Kolam Ikan Koi Menggunakan Metode Fuzzy Tsukamoto" 1: 14
- [2] Lintang, Elba, and Firdaus Firdaus. 2017. "Sistem Monitoring Kualitas Air Pada Kolam Ikan Berbasis Wireless Sensor Network Menggunakan Komunikasi Zigbee." In *Seminar Nasional Teknologi Dan Informatika 2017*. Muria Kudus University.
- [3] Pramana, Rozeff. 2018. "Jurnal Sustainable : Jurnal Hasil Penelitian Dan Industri Terapan Perancangan Sistem Kontrol Dan Monitoring Kualitas Air Dan Suhu Air Pada Kolam Budidaya Ikan" 07 (01)
- [4] Ifriza, Yahya Nur, and Djuniadi Djuniadi. 2015. "Perancangan Sistem Pakar Penyuluh Diagnosa Hama Padi Dengan Metode Forward Chaining." *Jurnal Teknik Elektro* 7 (1): 30–36
- [5] Wahyono, Irawan Dwi, and Mochammad Bagus Priyantono. 2020. "Implementasi Smart Home Pada Pendeteksi Dini Kebakaran Menggunakan Forward Chaining." *JITCE (Journal of Information Technology and Computer Engineering)* 4 (02): 63–71. <https://doi.org/10.25077/jitce.4.02.63-71.2020>.
- [6] Erik Hendra, Kurniawan. 2018. "Rule Based Expert System Pada Aplikasi Travel Asisten Menggunakan Metode *Forward Chaining* Berbasis Android." Universitas Islam Negeri Maulana Malik Ibrahim Malang